

# A Model to Evaluate Ray Tracing Hierarchies

K. R. Subramanian

Donald Fussell

Department of Computer Sciences

The University of Texas at Austin

## 1 Introduction

In this article we develop a model to evaluate ray tracing hierarchies. This model gives us a better look at the way hierarchies work in making ray tracing a tractable rendering technique. Some of the most important ray tracing hierarchies including the octree, bsp, median-cut, bounding volume hierarchies and the 5D space hierarchy fit into this model. Such a unified model will be able to tell us the underlying parameters that influence the performance of any ray tracing method. It also opens up the possibility of designing the hierarchy to improve performance.

## 2 The Model

Figure 0 shows a two level hierarchy. The proposed cost model is as follows:

$$\begin{aligned} Cost(T_A) = & \{Cost(A) + \alpha_A Cost(T_B) + \beta_A Cost(T_C) + \\ & (1 - \alpha_A - \beta_A)\{\gamma_A f_1(Cost(T_B), Cost(T_C), P_B) + \\ & (1 - \gamma_A) f_2(Cost(T_B), Cost(T_C), P_C)\}\} P_{A|A} \end{aligned}$$

where

- $T_A$  = subtree of node A
- $\alpha_A$  = probability that a ray segment is in the left region.
- $\beta_A$  = probability that a ray segment is in the right region.
- $\gamma_A$  = probability that a ray segment is in the right region.
- $f_1, f_2$  = terms that depend on the ray tracing hierarchy used.
- $P_A$  = probability that a ray intersects a primitive in A's subtree.
- $n_f$  = Bounding volume intersection cost.
- $P_{I|J}$  = probability that a ray that enters node I given that it enters J.

The proposed cost model computes the cost of a hierarchy recursively. It consists of three recursive terms and a non-recursive term.

## 2.1 Non-Recursive Term

This term accounts for the work done by each ray segment at each node before dealing with its children.

## 2.2 Recursive Term

This is a weighted sum of three terms, the weights(probabilities) being determined by the ray distribution. There are three possibilities; the ray segment might visit only the left child or the right child or both.  $\alpha_A$ ,  $\beta_A$  and  $(1 - \alpha_A - \beta_A)$  correspond to the respective probabilities. These probabilities respectively weight the costs of the left, right and both subtrees. Further, the coefficient of  $(1 - \alpha_A - \beta_A)$ , representing the cost of rays crossing the partition is a weighted sum of two terms, distinguishing the region of origin of each ray.  $\gamma_A$  and  $(1 - \gamma_A)$  represent the probabilities that any ray originates in the left or right region.  $P_B$ ,  $P_C$  are probabilities that a ray segment intersects a primitive in their subtrees and are needed in the cost computation. Lastly, the entire cost expression is multiplied by a conditional probability. At the root level, this is 1 since we are only considering rays that

intersect the bounding volume of the scene. This is not true at lower levels, for eg., the cost of  $T_D$  must be multiplied by  $P_{B|A_L}$ , which is the probability that rays enter the  $B$  given they enter  $A_L$ , the left region of  $A$ .

To illustrate this, the costs for  $T_B$  and  $T_C$  are as follows.

$$\begin{aligned}
Cost(T_B) &= \{Cost(B) + \alpha_B Cost(T_D) + \beta_B Cost(T_E) + \\
&\quad (1 - \alpha_B - \beta_B)\{\gamma_B f_1(Cost(T_D), Cost(T_E), P_D) + \\
&\quad (1 - \gamma_B) f_2(Cost(T_D), Cost(T_E), P_E)\}\} P_{B|A_L} \\
Cost(T_C) &= \{Cost(C) + \alpha_C Cost(T_F) + \beta_C Cost(T_G) + \\
&\quad (1 - \alpha_C - \beta_C)\{\gamma_C f_1(Cost(T_F), Cost(T_G), P_F) + \\
&\quad (1 - \gamma_C) f_2(Cost(T_F), Cost(T_G), P_G)\}\} P_{B|A_R}
\end{aligned}$$

We now proceed to apply this model to different ray tracing hierarchies.

### 3 Applying the Cost Model

#### 3.1 Bounding Volume Hierarchies

In bounding volume hierarchies, there are no separating planes chopping up space. To accommodate this in our model, we recognize that we visit both sides of the hierarchy when we compute the cost. In other words,  $\alpha_I = \beta_I = 0$  for any node  $I$ . Also,  $P_I = 0$  for any node  $I$ . There are no separating planes, so  $I_L = I_R = I$ .  $Cost(I) = n_f$ , a bounding volume test.

For a bounding volume hierarchy,

$$\begin{aligned}
f_1(., ., ., .) &= Cost(T_B) + (1 - P_B) Cost(T_C) \\
f_2(., ., ., .) &= Cost(T_C) + (1 - P_C) Cost(T_B)
\end{aligned}$$

and

$$\begin{aligned}
Cost(T_A) &= n_f + Cost(T_B) + Cost(T_C) \\
&= n_f + \{n_f + Cost(T_D) + Cost(T_E)\}P(B|A) + \\
&\quad \{n_f + Cost(T_F) + Cost(T_G)\}P(C|A)
\end{aligned}$$

### 3.2 Octree, BSP Hierarchies

The octree partitions space simultaneously along all three dimensions [Glassner 84]. The BSP tree hierarchy [Kaplan] does essentially the same but is implemented as a binary tree, partitioning along X, Y and Z and in that order. So, for purposes of cost analysis, these two structures are identical and will result in the same cost.

For a BSP or octree hierarchy,

$$\begin{aligned}
f_1(.,.,.,.) &= Cost(T_B) + (1 - P_B)(n_f + Cost(T_C)) \\
f_2(.,.,.,.) &= Cost(T_C) + (1 - P_C)(n_f + Cost(T_B))
\end{aligned}$$

and

$$\begin{aligned}
Cost(T_A) &= 1_{cmp} + \alpha_A Cost(T_B) + \beta_A Cost(T_C) + \\
&\quad (1 - \alpha_A - \beta_A)\{\gamma_A(Cost(T_B) + (1 - P_B)(n_f + Cost(T_C))) + \\
&\quad (1 - \gamma_A)(Cost(T_C) + (1 - P_C)(n_f + Cost(T_B)))\}
\end{aligned}$$

All the non-recursive terms can be lumped into  $Cost(A)$

$$\begin{aligned}
Cost(T_A) &= \{1_{cmp} + \{\gamma_A(1 - P_B)n_f + (1 - \gamma_A)n_f\}_{int} + \\
&\quad \alpha_A Cost(T_B) + \beta_A Cost(T_C) + \\
&\quad (1 - \alpha_A - \beta_A)\{\gamma_A(Cost(T_B) + (1 - P_B)(Cost(T_C))) + \\
&\quad (1 - \gamma_A)(Cost(T_C) + (1 - P_C)(Cost(T_B)))\}\}P(A|A)
\end{aligned}$$

where

$$Cost(A) = 1_{cmp} + \{\gamma_A(1 - P_B)n_f + (1 - \gamma_A)n_f\}_{int}$$

The expression for  $T_B$  is similar but has a conditional probability in addition.

$$\begin{aligned} Cost(T_B) = & \{Cost(B) + \alpha_B Cost(T_D) + \beta_B Cost(T_E) + \\ & (1 - \alpha_B - \beta_B)\{\gamma_B(Cost(T_D) + (1 - P_D)(Cost(T_E))) + \\ & (1 - \gamma_B)(Cost(T_E) + (1 - P_E)(Cost(T_D)))\}\}P(B|A_L) \end{aligned}$$

and so on.

### 3.3 $K - d$ tree Hierarchy

This is a binary hierarchy using separating planes orthogonal to the X,Y and Z axes but has important differences from the BSP hierarchy[Kaplan]. The hierarchy is balanced on the number of objects and the planes are chosen to produce this kind of partitioning. The number of objects straddling the plane is also minimized. There is considerable flexibility in where a separating plane can be positioned and we have three dimensions to choose from. The traversal algorithm makes it unnecessary for the convex subsets that are produced by the space partitioning to be contiguous.

In this scheme, a bounding volume test and a separating plane test is done before we investigate a node's children.

So the node cost is

$$Cost(A) = (1 + n_f)_{int}$$

and

$$\begin{aligned} Cost(T_A) = & \{\{1 + n_f\}_{int} + \\ & \alpha_A Cost(T_B) + \beta_A Cost(T_C) + \\ & (1 - \alpha_A - \beta_A)\{\gamma_A(Cost(T_B) + (1 - P_B)(Cost(T_C))) + \\ & (1 - \gamma_A)(Cost(T_C) + (1 - P_C)(Cost(T_B)))\}\}P(A|A) \end{aligned}$$

and

$$\begin{aligned}
Cost(T_B) = & \{Cost(B) + \alpha_B Cost(T_D) + \beta_B Cost(T_E) + \\
& (1 - \alpha_B - \beta_B)\{\gamma_B(Cost(T_D) + (1 - P_D)(Cost(T_E))) + \\
& (1 - \gamma_A)(Cost(T_E) + (1 - P_E)(Cost(T_D)))\}\}P(B|A_L)
\end{aligned}$$

### 3.4 Spatial Enumeration method - ARTS

In this method, the partitioning is uniform along all 3 dimensions. Its identical to an octree or BSP partitioning except that the partitioning is not adaptive. Thus the only difference in the form of the cost expression is the node cost, which will be replaced by the voxel hopping cost. As this method uses an extended Bresenham's line algorithm (3DDDA) in three dimensions, the cost will be much smaller since this is an incremental algorithm.

### 3.5 5D Space method - Arvo

In this method, space is partitioned over the three dimensions just like the octree. In addition, the two directional dimensions of the ray is also partitioned creating *beams* in 5-space. The partitioning is again adaptive. The leaf nodes of the hierarchy are beams of hypercubes. A ray belongs to exactly one hypercube in the hierarchy. Thus, given a ray, we trace out a path from the top of the hierarchy down to a beam where an ordered search is conducted to determine the closest intersection.

For our cost analysis of the 5-D hierarchy we can consider this structure as a binary tree with the partitioning going through X,Y,Z,U,V dimensions in sequence (the last two are the directional dimensions). However, we always go only on one side of the binary tree at any node. Thus, in our model,  $\alpha_I = 1, \beta_I = 0$ , or  $\alpha_I = 0, \beta_I = 1$  for any node  $I$ .  $(1 - \alpha_I - \beta_I) = 0$  so its coefficient is irrelevant in the cost expression. Node cost is just 1 compare and there are no bounding volume tests in the hierarchy.

Thus the cost is

$$Cost(T_A) = \{1_{cmp} + Cost(T_B)\}P(A|A)$$

or

$$Cost(T_A) = \{1_{cmp} + Cost(T_C)\}P(A|A)$$

and

$$Cost(T_B) = \{1_{cmp} + Cost(T_D)\}P(B|A_L)$$

or

$$Cost(T_B) = \{1_{cmp} + Cost(T_E)\}P(B|A_R)$$